National College of Ireland

# Defensive AI for Customer Service Chatbots: Detecting and Mitigating Adversarial Prompt Injections

MSCCYBETOP

Alan Boyce
Student ID: X23299517

School of Computing
National College of Ireland

Supervisor: Raza Ul Mustafa

**National College of Ireland**

**MSc Project Submission Sheet**

**School of Computing**

**Student Name:** Alan Boyce……

**Student ID:** X23299517……..……

**Programme:** MSCCYBETOP…… **Year:** 2025……..

**Module:** ……..……

**Supervisor:** Raza Ul Mustafa……..……

**Submission Due Date:** 15/09/2025……..……

**Project Title:** Defensive AI for Customer Service Chatbots: Detecting and Mitigating Adversarial Prompt Injections……

**Word Count:** 9737…… **Page Count 23**……..……..

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

**Signature:**  ……..……

**Date:** 15/09/2025……

**PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST**

| | |
|---|---|
| Attach a completed copy of this sheet to each project (including multiple copies) | □ |
| **Attach a Moodle submission receipt of the online project submission,** to each project (including multiple copies). | □ |
| **You must ensure that you retain a HARD COPY of the project**, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer. | □ |

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

| **Office Use Only** | |
|---|---|
| Signature: | |
| Date: | |
| Penalty Applied (if applicable): | |

# Defensive AI for Customer Service Chatbots: Detecting and Mitigating Adversarial Prompt Injections

Alan Boyce
X23299517

**Abstract**

The rise of Large Language Models (LLMs) in customer service chatbots has created new opportunities for faster, more scalable support while introducing new vulnerabilities, particularly in the form of prompt injection attacks. These attacks exploit the LLMs, which are designed to understand and respond to virtually any input. This allows malicious users to bypass restrictions or expose confidential data from the system. Although many defensive strategies have been proposed, most remain static and fail to adapt to evolving threats or multi-turn adversarial patterns.

This research proposes a proof-of-concept hybrid adversarial detection system that combines a fine-tuned RoBERTa classifier alongside a session-aware Graph Neural Network (GNN) capable of modelling conversational context. The system also includes a logging and feedback loop to support retraining over time, adapting to emerging threats. The chatbot utilises the Mistral-7B model, accessed through a Streamlit app frontend and backed by an SQLite database for logging interactions.

Evaluation was conducted using a structured test suite alongside real-world Man-In-The-Middle (MITM) simulation attacks. The baseline chatbot, with minimal defences, displayed detection accuracy of 52.00%, failing to block a significant number of adversarial inputs. The classifier alone improved accuracy to 87.00%, while the complete hybrid system saw that figure rise to 95.00% in both structured testing and real-world scenarios. These results highlight the effectiveness of combining static and contextual reasoning in adversarial detection.

As a proof-of-concept, this work demonstrates the viability and efficacy of adaptive, multi-layered defences in protecting LLM-powered chatbots and lays a foundation for future research into self-improving, context-aware systems.

# 1 Introduction

The adoption of LLMs into customer service chatbots has transformed how businesses interact with customers, from managing queries, automating support and enhancing user experience. As with any new technology, this integration has introduced new security risks, particularly prompt injection attacks, where malicious users manipulate input to alter the model behaviour, subvert restrictions or leak sensitive information. These interactions are a serious concern for customer service environments where interactions may involve sensitive personal information.

Prompt injection attacks can be direct, using skilfully crafted inputs or indirect where malicious instructions may be embedded in external content that the chatbot is instructed to process. Despite the evolving nature of these attacks, existing defences have failed to keep pace. Most defences rely on static rules, prompt filtering or fine-tuning models, methods which work on known threats but struggle on subtle or obfuscated multi-turn attacks. These limitations can be observed in customer

service chatbots where conversations can unfold across turns and adversarial behaviour may develop subtly over time.

This research proposes to address these challenges by designing and evaluating a proof-of-concept hybrid adversarial detection system that combines a fine-tuned RoBERTa classifier with a session-aware Graph Neural Network (GNN). The GNN is used to model prompt interactions across a session, capturing both temporal and semantic patterns that infer evolving adversarial behaviours. This system also introduces a feedback loop that enables retraining on newly detected attack patterns ensuring a defensive system that is adaptable and scalable.

The overall aim of this work is to evaluate the effectiveness of a hybrid approach to detecting and mitigating adversarial prompt injection attacks in customer service chatbots. While initially scoped as a proof of concept, this system is designed with future extensibility and demonstrates how combining prompt level and context-aware reasoning can lead to a much more resilient system. The report proceeds as follows, Section 2 reviews existing literature, Section 3 outlines the design methodology, Section 4 details the implementation, Section 5 presents evaluation results and Section 6 offers conclusions and directions for possible future work.

# 2 Literature Review

Large Language Models (LLMs), such as OpenAI's ChatGPT, have seen their integration with real-world applications, in particular customer service chatbots, transform how businesses interact with customers and access information. These chatbots, leveraging LLMs provide fast, natural human sounding support across a wide range of industries. Even though these systems are convenient and scalable, their use has created a new class of security vulnerabilities, with prompt injection attacks becoming a significant risk.

Prompt injection can occur in two primary ways: directly, through placing deceptive comments into a user's message or indirectly, by embedding those commands within external content that the model later interprets. Although model alignment and safety bypasses were initially discussed in relation to these attacks, more recent research is increasingly placing them within established cybersecurity frameworks. Applying the CIA Triad – Confidentiality, Integrity, and Availability, to this type of attack, demonstrates how such vulnerabilities can impact the accuracy of these models, have the unintended purpose of leaking sensitive information and impair the system's functionality (Jones et al., 2025). These issues are quite concerning in customer service environments where customer interactions can involve divulging sensitive information.

A growing body of research has sought to understand, categorise and defend against prompt injection attacks (Liu, 2024; Muliarevych, 2024). Included in the research are taxonomies of attack vectors, real-world system evaluation to identify vulnerabilities and a variety of proposed defences, such as input filtering and adversarial training (Liu, 2024; Muliarevych, 2024). Yet, despite their value, many of these defences share a similar shortcoming, they are inherently static. If based on input defined rules, carefully composed prompts or models which have been retrained, these approaches lack the flexibility to keep pace with the ever-evolving nature of these threats. As attackers refine their techniques, these static defences are inclined to adapt less swiftly or effectively enough to offer adequate protection (Liu, 2024; Muliarevych, 2024).

Additionally, current LLMs have not been designed to notice or interpret how a user's behaviour or language changes over time, especially over multi-turn interactions. Malicious prompt detection typically occurs at the input/output level. It can often neglect wider patterns in conversational flows, previous exchanges or multi-turn adversarial dialogue patterns (Kulkarni et al., 2025). In customer service chatbots in particular, this can reduce the effectiveness of detection systems where malicious prompts may be woven into seemingly legitimate conversations.

This literature review examines the current landscape of prompt injection research, covering core areas such as detection methods, mechanisms for defence and classification systems. Even with significant progress having been made in understanding and applying mitigation strategies against these attacks, a persistent gap remains in the development of adaptive, context-sensitive trained defensive models. To close this gap, this research draws on techniques from another field, Graph Neural Networks (GNNs) which have been applied to Intrusion Detection Systems (IDS), malware detection and identifying Advanced persistent Threats (APTs). By tracking prompt behaviour over time and incorporating an active feedback loop, GNNs could lay the groundwork for a new generation of defences, one capable of evolving alongside adversarial attacks and thus safeguard these LLM-driven customer service chatbots.

## 2.1 Existing Detection and Defence Mechanisms

With the awareness of prompt injection attacks have grown in recent years so too has the research into addressing and mitigating them. Current approaches to defending against these attacks broadly fall into two categories: **static defences**, those that aim to prevent known attack patterns using predefined rules or constraints on the model (Muliarevych, 2024; Tshimula et al., 2024) and **formalisation,** which seeks to better understand and categorise threats or attack vectors through system analysis and benchmarking (Liu, 2024; Yi et al., 2023). While both have contributed significantly to the early days of prompt injection attack mitigation strategies, they both display limitations, particularly when we discuss their adaptability and their robustness in the long-term (Charfeddine et al., 2024).

### 2.1.1 Static Defences

Static defences provided some of the earliest strategies in defending against prompt injection attacks. Examples include input sanitisation, prompt rewriting, hard coded restraints and content filtering to restrict the output generation of chatbots (Muliarevych, 2024). Some systems rely upon datasets of known adversarial attacks, or specifically curated ones to fine-tune LLMs and thus increase their resilience to these specific patterns of attack. Others rely on architectural workarounds such as placing the LLM's behind a control layer filtering the inputs and outputs before they can be processed further (Tshimula et al., 2024).

Although these defences can be quite effective against known or simplistic attacks, they are intrinsically frail. Systems that use a rule-based dataset can be vulnerable to evasion through obfuscation and direct/indirect paraphrasing. Additionally, static defence can fail to allow for contextual shifts, they treat inputs in isolation rather than as longer multi-turn conversations, which may leave them blind to subtle malicious shifts in inputs over time.

In customer service chatbots this weakness may be intensified by the assortment and randomness of user inputs. A prompt that starts out benign could become malicious when combined with earlier or later inputs, a use case that static defences currently are not equipped to deal with.

### 2.1.2 Formalisations and benchmarking

Alongside implementation-based defences, researchers have worked to develop structured taxonomies and evaluation frameworks to better understand prompt injection. Notably, Rossi (2024) introduced an early classification system that organised prompt injection attacks by input type and injection methods. Building on that, Liu (2024) formalised the broader landscape of prompt injection threats and introduced benchmarking criteria to effectively assess various defensive approaches. This research offers important conceptual clarification and lays the groundwork for future, more thorough threat modelling.

However, this research often stops short of proposing adaptive solutions. While they help to define the threat landscape, they do not outline the tools required to defend against ever evolving attacks in real-time. They are essentially maps without motion; they define the landscape but fail to account for the change in that landscape with innovative prompt injection attacks or tactics.

Research has been undertaken to determine more creative or adversarial strategies, such as the idea of ‘hacking back’, embedding counter-prompts into input flows or using adversarial training to increase the robustness of the model (Iqbal et al., 2023). While these techniques show promise they remain largely reactive and manually curated resulting in limitations in scalability and practical application in fast paced, high volume environments such as customer service chatbots.

### 2.1.3 The missing element: Adaptability and feedback

Both static and formalised approaches when considered display a shared limitation, a lack of real-time adaptive feedback mechanisms. Most defence strategies serve as a single-shot event to be detected or blocked at the time of input. Little capacity has been given to observe patterns over time, evolving conversational flows, or the incorporation of additional knowledge learned from previous attacks into new future defences.

This highlights a critical gap particularly in adversarial settings where attackers can adjust their employed strategies in response to the defences that have been deployed. In traditional cybersecurity, these problems have been addressed using dynamic threat detection systems that evolve alongside the dynamic threat landscape. A comparable evolution is required in the LLM/customer service chatbot space.

The next section details how this problem can be addressed through the novel use of architecture, that is by adapting Graph Neural Networks from the field of malware and Advanced Persistent Threats detection to model user chatbot interactions as evolving, structured data. This approach offers an intriguing entry to enable adaptive, context-aware defences which are capable of operating in real-world customer service chatbot environments.

## 2.2 Limitations in current approaches

Despite the growing body of research focused on detecting and mitigating prompt injection attacks, current approaches demonstrate several limitations. When existing defences are evaluated against the current dynamic, evolving nature of adversarial techniques, especially in customer service chatbots where inputs are open ended, unstructured and unpredictable, these weaknesses become apparent.

Major challenges exist in defending against prompt injection attacks particularly those that are static in nature. Techniques such as input filtering, output restriction, prompt hardening and rule-based hardening are built around predefined patterns of known attack vectors (Muliarevych, 2024; Tshimula et al., 2024). While these techniques can work well in controlled environments or against known adversaries, they frequently fail in the face of more subtle, indirect attacks, especially those that develop over the course of multi-turn dialogues. As shown by Yu et al. (2023), many deployed custom GPTs remain vulnerable to such attacks despite deployed safeguards, highlighting the limitation of static filtering mechanisms.

Even where formal frameworks are present, such as the categorisation by Rossi (2024) or the benchmarking conventions introduced by Liu (2024), the focus remains on static one-time assessments. These methods provide essential insights into the nature and structure of prompt injection attacks, but they lack the facility to adapt dynamically as attacker's strategies evolve. Similarly, Pasquini et al. (2024) proposed an innovative approach using counter-prompts as a defence strategy, however this approach still relies on manual oversight and predefined attack-response pairs, further hindering the scalability and long-term adaptability.

The practical challenges of deploying LLMs in real-world customer service environments are often ignored. In practice, these models engage in high frequency, high variance interactions with users who may be unfamiliar with or unintentionally testing the limits of the system. The volume and unpredictable nature of these conversations make it impractical to solely rely on static defences such as input filtering or rule-based defences. What is required is a dynamic defence framework that can learn from prior interactions, identify suspicious patterns and continuously refine its detection strategies over time.

These limitations point to a clear gap in the literature, the absence of adaptive, feedback driven systems for detecting prompt injection attacks in LLM deployed customer service chatbots. Unlike static defences, such a system would monitor user behaviour, track conversational flows, and analyse input/output patterns over time, much like the kind of temporal and structural modelling used in cybersecurity in APT detection and intrusion analysis (Bahar, 2025; Qiao et al., 2024; Zhong, 2024).

## 2.3 Towards Adaptive detection: Insights from Graph Neural Networks.

The shortcomings of current prompt injection defences underscore the need for a fundamentally different approach, one that adapts alongside the threat landscape rather than being reactive to it (Liu, 2024; Pasquini et al., 2024). Within the cybersecurity landscape this adaptability has been achieved through graph-based threat detection with Graph Neural Networks (GNNs) playing a key role (Zhong, 2024). These models have been proven to be effective in malware classification and APT tracking

where the nature of these attacks is inherently multistage, context sensitive and dynamic, traits that can be seen to be increasingly shared with prompt injection attacks.

GNNs are designed to process graph-structured data making them uniquely capable of capturing relationships and dependencies that unfold over time and across systems. For example, in traditional Intrusion Detection Systems (IDS) GNNs are used to model network flows, user interactions and system level events as interconnected graphs where nodes can represent elements such as IP addresses to executed commands and edges represent temporal or casual relationships (Zhong, 2024; Bahar, 2025). This structure enables the identification of subtle long-range patterns that static or rule-based models can miss.

The structural and temporal patterns observed in chatbot patterns mirror those found in IDS and APT use cases. A typical exchange between user and a chatbot unfolds as a temporally ordered sequenced set of messages comprising of both user queries and model responses which can be modelled as a graph. Nodes correspond to the messages and the edges capture the direction of the conversational flow or contextual references. Prompt injection attacks, particularly those that are indirect or multi-turn naturally embed themselves within the conversational structure making graph-based modelling a promising direction for detection.

GNN techniques used for malware and APT detection can be adapted to use within the security context of LLM-driven customer service chatbots.

- Temporal GNNs like CONTINUUM (Bahar, 2025), model sequential event chains, making them capable of detecting delayed or cumulative manipulations, features of indirect prompt injection attacks.
- Provenance aware frameworks such as SLOT (Qiao et al., 2024) and P3GNN (Nazari et al., 2024) trace how information or control flows through a system, which lends itself well to track adversarial instructions as they influence model behaviour downstream.
- Explainable GNNs like those used in malware detection (Mohammadian et al., 2024) offer critical insights behind the classification decisions, an essential feature in maintaining transparency and trust in live chatbot deployments.

In contrast to static defences, GNNs offer the capability to adapt over time by adjusting detection thresholds and including feedback from historical interactions. Within the context of a customer service chatbot this could mean recognising recurring prompt patterns, signs of escalation, or shifts in the context of conversations that could indicate a developing prompt injection attack. Gradually, the model can construct a behavioural graph profile of users and flag anomalous activity without relying on handcrafted rules.

This thesis proposes the integration of a GNN driven feedback mechanism into the prompt injection pipeline for customer service chatbots. By modelling conversation history as a dynamic graph and updating it as the conversation evolves, the system can continuously assess incoming prompts in light of their evolving context. This leverages advantages that static or rule-based systems lack, allowing for the early detection of developing risks and improved resilience to novel attack routes.

In doing so this research connects two domains, the emerging field of LLM prompt security and the well-established techniques of graph-based detection within the cybersecurity landscape. The

following section details the proposed architecture, outlines the evaluation criteria and expected contributions.

## 2.4 Research gap and proposed contribution

Despite the focused and growing attention on prompt injection attacks in LLMs, particularly in high-risk applications such as customer service chatbots there remains a significant gap in the literature regarding adaptive, context-aware detection methods. As previously discussed in earlier sections most defences rely upon static, prompt filters or one-shot adversarial training defences which are often lacking in countering evolving, multi-turn attack strategies (Muliarevych, 2024; Tshimula et al., 2024). Even those that propose more formal frameworks such as Rossi (2024) and Liu (2024) typically fall short of proposing models capable of supporting real-time adaptability in these open-ended user interactions.

Additionally, while certain creative countermeasures have emerged such as offensively using prompt injections to neutralise malicious input (Pasquini et al., 2024) these remain largely manually crafted and can be difficult to scale in production environments where the customer service chatbots are expected to operate autonomously and continually. This exposes a critical vulnerability, the absence of an adaptive feedback-driven system that can learn from previous interactions, model conversational behaviour and continually improve detection rates over time.

To address this challenge, this thesis proposes a defensive AI model grounded in Graph Neural Networks (GNNs). Drawing inspiration from their effective use in dynamic detection systems in cybersecurity such as APT detection and malware classification (Zhong, 2024; Bahar, 2025; Qiao et al., 2024) this approach will treat chatbot interactions as dynamic graphs where each prompt, user interaction, and system responses forms a structured flow of interactions.

Using a GNN to model and update this interaction graph in real-time, the system can detect prompt injection attacks not as isolated anomalies but as part of a developing behavioural pattern. This allows for:

- Temporal awareness – capturing delayed or multi-turn attacks.
- Structural sensitivity – understanding how injected prompts could alter later responses.
- Adaptive learning – through gathered feedback and graph histories the model is continuously updated to improve its performance.

In doing so, this research offers several contributions:

1. It introduces a novel graph-based defence architecture for detecting adversarial prompt injection attacks in LLM-driven customer service chatbots.
2. It adapts established GNN techniques from the domain of cybersecurity for use in chatbot systems, promoting interdisciplinary application of AI in threat detection.
3. It evaluates this model effectiveness against static defences providing comparative insights aligning with the research question to provide measurable outcomes.

Ultimately, the aim of this research is to provide improved accuracy for prompt injection attacks while advancing the wider goal of adaptive AI safety in real-world natural language applications.

## 2.5 Summary

This literature review has explored the evolving threat of malicious prompt injection in the context of customer service chatbots. It began by framing these attacks not merely as operational vulnerabilities but as significant threats that could undermine the foundational pillars of the CIA triad, Confidentiality, Integrity and Availability (Jones et al., 2025). The review then assessed the shortcomings of current existing defences, which largely rely on static rule sets, input filtering or pretrained constraints that fail to adapt new or obfuscated attacks (Muliarevych, 2024; Tshimula et al., 2024).

While taxonomic efforts (Rossi, 2024) and benchmarking standards (Liu, 2024) have contributed to conceptual clarity, the existing research lacks the capacity to detect attacks as they occur in real-time conversational flow. Additionally, reactive solutions including prompts as an offensive measure (Pasquini et al., 2024) remain largely manual and lack scalability in terms of high volume, dynamic environments such as customer service chatbots.

In response to this, the thesis proposes a defensive AI architecture based on GNNs. Building on their successful implementation and use in cybersecurity applications such as APT detection and malware analysis (Bahar, 2025; Qiao et al., 2024), GNNs provide a framework to model conversation history, track prompt interactions over time and enables real-time risk assessment through a dynamic active feedback loop. This GNN based approach positions it as a significant evolution over static mechanisms, one that can identify prompt injection attacks not as single anomalies but as a behavioural pattern within a structured conversational flow.

By utilising the strengths of GNNs and aligning them with the complex nature of adversarial prompt injection attacks, this research introduces a novel, adaptive detection framework that directly addresses critical gaps identified in the current literature. In doing so, it directly responds to the central research question – How effective is a defensive AI model in detecting and mitigating adversarial prompt injection attacks in customer service chatbots?

# 3 Design Methodology

## 3.1 System Overview

The system designed in this research was developed to detect and mitigate a broad range of adversarial prompt injection attacks including those that purposefully attempt to manipulate chatbot behaviour, bypass restrictions or expose sensitive information within customer service chatbots. The underlying architecture adopts a multi-layered approach, which combines a fine-tuned RoBERTa classifier and a Graph Neural Network (GNN) operating within an active feedback loop. This hybrid approach enables the detection of adversarial behaviours in both single turn and multi-turn conversational flows. By analysing not only individual prompts but also the wider relational context in which these prompts appear, the success rate in detection is greatly enhanced. Furthermore, by integrating real live user interactions into a retraining process, the system supports dynamic learning from new interaction patterns over time boosting its detection capabilities and adaptation to evolving attack strategies.

## 3.2 System Objectives

The fundamental objective of this system is to provide real-time protection for customer service chatbots against adversarial prompt injections attacks which aim to exploit or bypass safety mechanisms. The system has been designed to be proactive, it pre-emptively identifies any malicious input before they can be processed by the chatbot, serving as a defensive gateway. In addition to immediate classification and blocking, the system persistently logs flagged prompts and their metadata in a lightweight SQLite database. This enables follow up analysis and manual or automated reviews. Beyond reactive defence, the system also looks to model semantic and temporal relationships between user interactions using graph structures, which are analysed by a GNN to identify the more subtle or persistent adversarial behaviour. These insights feed into a retraining process, ensuring the improvement of the system's detection capabilities improve iteratively over time.

## 3.3 Architecture Overview

The full pipeline comprises several interconnected components, each responsible for a specific function in the adversarial detection and response workflow. The chatbot itself is built on the Mistral-7B model, served locally using the Ollama platform. A carefully constructed system prompt instructs the chatbot to adopt a restricted customer service role handling enquiries relating to customer orders, deliveries and returns. The frontend has been developed using Streamlit and provides a customer support interface that simulates user queries within this domain.

At the input stage, all prompts are intercepted by a transformer-based classification layer powered by RoBERTa. This model has been fine-tuned on a mixed dataset of benign and adversarial inputs, including the Tensor Trust, Wild Jailbreak and In-the-Wild-Jailbreak datasets covering a broad range of jailbreak techniques and obfuscation strategies. RoBERTa's strength lies in dynamic masking, deeper and longer training and the exclusion of Next Sentence Prediction (NSP) which allows it to outperform the earlier BERT model in tasks involving nuanced language understanding. It classifies incoming prompts as either safe, potentially adversarial or adversarial based on its learned representations.

Prompts flagged by the classifier are recorded in a structured SQLite database, including the text, classifier confidence score, timestamp and the chatbot response which aligns with user input. This enables both long-term tracking of attacks and retrospective review of false positives or emerging patterns.

These logged interactions are subsequently used to build graph structures, where nodes represent prompts, responses, and classification decisions. While edges encode semantic similarity and temporal flow within sessions. The graph is analysed by a GNN implemented using PyTorch Geometric, which can detect suspicious clustering or propagation patterns—particularly in multi-turn attack strategies that may evade static defences.

Based on GNN findings and manual validation, particularly high-confidence, adversarial prompts are fed back into the training dataset. This feedback loop supports a continuous learning system in which the RoBERTa classifier is incrementally retrained to reflect real-world adversarial trends.

## 3.4 Workflow and Data Flow

The system is designed to be modular and showcase a staged decision flow (see Appendix A, Figure 1). When a user submits a prompt, it is firstly processed by the fine-tuned RoBERTa classifier. If the

input is deemed to be benign, the prompt is forwarded to the chatbot which generates a response as normal. However, if the classifier deems the prompt to be adversarial or potentially so, the prompt is blocked from reaching the chatbot and is logged to the database along with its metadata. Any prompt flagged as potentially adversarial by the classifier is further scrutinised by the GNN in context. Using graph structures, the GNN can detect subtler multi-turn or obfuscated adversarial prompts. If the GNN interprets a prompt as adversarial, it logs the malicious prompt to the database along with its metadata.

All flagged instances are stored in a feedback database along with rich metadata. This dataset represents the evolving set of suspected adversarial prompts. At defined intervals, and once a threshold of new data has accumulated, the curated database can be utilised to rebuild the graph structures, by incorporating both old and new adversarial prompts. As before, nodes represent the prompts, while the edges encode the semantic and temporal relationships.

Once the graphs have been rebuilt, the GNN will process the data to uncover relational patterns which indicate adversarial behaviour. Where such patterns are identified, the relevant prompts can be selected for inclusion in the retraining dataset. This approach allows the system to function not only as a static filter but also as a dynamic, learning driven defence framework against adversarial attack.
From this analysis a subset of prompts is selected for inclusion in the RoBERTa retraining dataset. This includes new types of attacks unearthed by the GNN, or variants that slipped past RoBERTa but were identified in context. The RoBERTa model is then further fine-tuned using this curated data improving its ability to detect previously missed adversarial prompts while reducing reliance on the GNN for more common cases.

The process is entirely supported by a robust testing strategy, in which both real and synthetic adversarial prompts are used. Baseline evaluations are underpinned by an approach whereby an unprotected version of the chatbot is tested to understand its vulnerability against benign and adversarial prompts. These tests are repeated once the required adversarial defences have been implemented to understand and measure the system's improvement over time. Real-world attacks, such as prompt injection and context hijacking are simulated using tools such as BurpSuite to further demonstrate the effectiveness of the new defence measures.

### 3.4.1 Data Protection and GDPR Considerations

Data collection is an essential component of this study which focuses on retaining prompts, classifier outputs and session metadata. However, this type of information can raise privacy and confidentiality issues in real world applications. The system has been deliberately designed to exclude the storing of any Personally Identifiable Information (PII). It captures the essential components such as the user prompt, prediction label, timestamp and session IDs. These records are maintained locally in an SQLite database with no cloud or external connections ensuring full control over the data as well as traceability of the data.

In a real-world deployment, stronger safeguards would be necessary. Examples include anonymisation of the collected data, hashing of session identifiers, applying data retention rules, and providing an opt-out option to users. Applying these measures would help ensure compliance with the General Data Protection Regulation (GDPR) including data minimisation, storage limitations, integrity and confidentiality, and purpose limitations. Informing users of the logging process maintains transparency and accountability presented in line with GDPR's principles of lawfulness,

fairness and transparency. All of this combined, ensures that any logging activity undertaken respects user privacy and regulatory obligations.

## 3.5 Design Rationale

The design choices made in this research are underpinned by what was firstly practical and achievable with the current tools and resources available and a desire to go beyond existing methods to try for something more innovative in improving security in customer facing chatbots. The motivation behind the selection of RoBERTa was driven by the knowledge that the classifier was trained on a much larger dataset and for longer than BERT, while it removed the Next Sentence Prediction (NSP) which helped the performance of the classifier. RoBERTa utilises dynamic masking improving the generalisation of the model.

Logging with SQLite was chosen as there was a requirement for a simple, lightweight, fast and reliable solution for local logging making it quite useful for iterative experimentation.

The decision to incorporate GNNs into the workflow stems from the understanding that adversarial behaviours are often better understood in context particularly where conversations can occur across multi-turn interactions or where patterns are repeated over time. Unlike classifier models that look at the prompt without context, the GNN can in contrast spot suspicious behaviour by detecting unusual patterns in how prompts are connected while identifying subtle changes in meaning or tone over time.

Finally, the active feedback loop represents a key innovation in that it does not treat the classifier as a fixed component; it is retrained as new data becomes available, creating a dynamic defence mechanism that evolves over time as new attack strategies emerge. This reflects a defence that is adaptive and ultimately effective against persistent and creative adversarial attacks.

# 4 Implementation

## 4.1 Chatbot Deployment

The Mistral 7B model powers the chatbot which is hosted locally via the Ollama platform. This setup favours performance and offers more control over the model in an environment constrained by resources. This model was selected due to its balance of natural language generation and its resource efficiency, making it reliable when it comes to real-time, natural customer interactions. The model's frontend is contained within a Streamlit application, allowing the user to interact with the customer service chatbot through a simple responsive web page as a user would experience in the real-world.

A carefully constructed system prompt ensures that the chatbot retains a helpful, professional, polite tone as expected of a real-world customer facing chatbot. This frontend-backend integration presents a simple chatbot interface masking the complexity of the underlying adversarial detection system.

## 4.2 Classifier Fine-Tuning

At the heart of the detection system is a RoBERTa-base model fine-tuned on a curated dataset of both benign and adversarial prompts, chosen for its robust performance in natural language and contextual understanding and classification tasks. As discussed in section 3.5 Decision Rationale, RoBERTa's architecture allows it to capture contextual nuance, an essential capability when distinguishing between benign and adversarial prompts.

The classifier was trained on an amalgamated dataset that includes both benign and adversarial inputs. Legitimate prompts were sourced from the Twitter Customer Support (TWCS) dataset and the Dialogue State Tracking Challenge (DSTC) dataset. Filtering was applied to each dataset to ensure that all benign samples used aligned with the chatbot's current role and domain of customer orders, deliveries and returns. Adversarial prompts were pulled together from multiple sources including the Tensor Trust dataset, Wild Jailbreak, In-the-Wild-Jailbreak datasets alongside a curated sample of custom designed prompts to simulate real world attack behaviours. These included prompt injections, jailbreak attempts, context hijacking, indirect attacks and subtle obfuscation strategies.

Before the model was trained it required preprocessing which involved tokenising the dataset, truncating the input to a sequence length of max 256 tokens and then encoded with attention masks. The classifier was trained using the Hugging Face Trainer API. The performance of the training was tracked using Weights & Biases, a tool used to monitor and visualise training progress using key metrics such as accuracy, precision, recall and F1-score. Each metric was logged across each epoch of training providing insights into how well the model was learning or if it was starting to overfit. Initial evaluation results required the retraining of the classifier on an augmented dataset that included a subset of prompts misclassified as false positives. This refinement step aimed to reduce false positives in real-world testing.

Using a Lenovo LOQ laptop equipped with 16 GB RAM, an AMD Ryzen 7 CPU, an Nvidia RTX 4070 GPU and a 1TB SSD, training of the classifier was completed in ~25 hours. This training involved preprocessing of the combined datasets, fine-tuning the model over three epochs and evaluation. This duration reflects the resource-intensive nature of transformer-based models while remaining feasible within the constraints of a local workstation setup.

The final model produced a classification of either benign or adversarial for each input while providing a confidence score which enables threshold-based decision making and fine-tuning during evaluation and system testing.

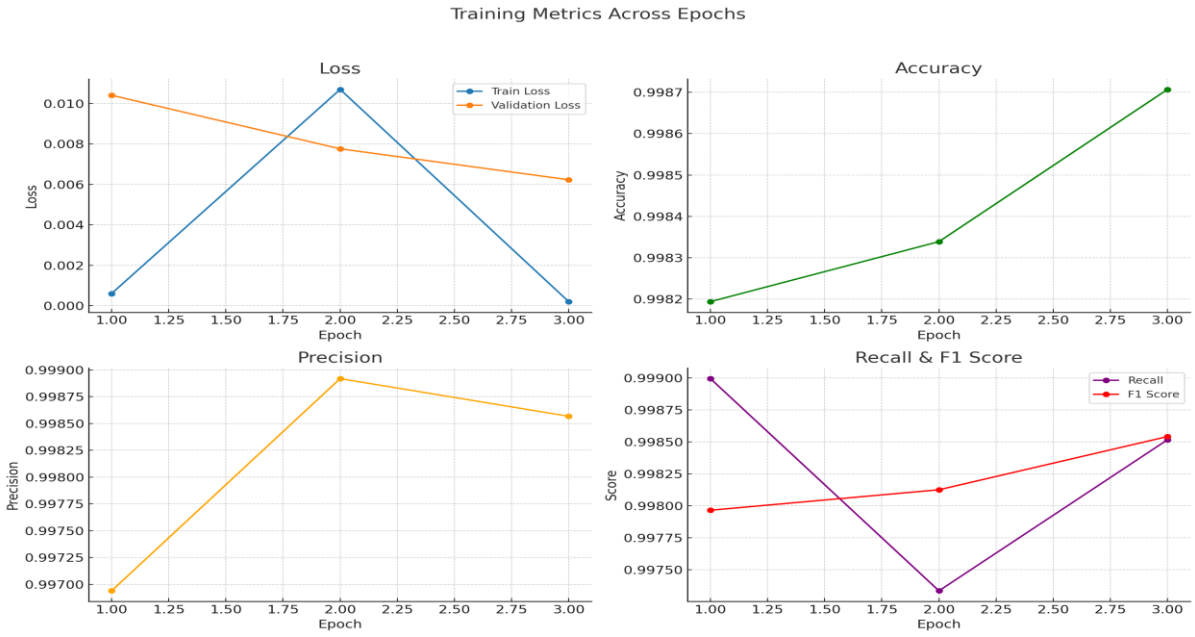


**Figure 1: Training metrics across Epochs**

## 4.3 Prompt Logging System

To support auditing, analysis and feedback driven learning, all prompts whether classified as safe, potential or adversarial are persistently logged to an SQLite database. This lightweight database provides a low-cost, efficient way of logging all prompts and their metadata. Each record added to the database contains the original prompt, the classifier's prediction label, confidence score, timestamp and the session identifier so multiple prompts from the same session can be linked. If applicable, the chatbot response to the adversarial prompt is stored.

This logging mechanism is a critical part of the system's memory, it allows for reviews after the fact, of misclassified or low confidence scored prompts. Logged prompts become an important training asset, they can at a later stage be manually annotated, re-evaluated and together with session graphs now saved after each conversation can be used for GNN retraining. As a result, the database and stored graphs not only enable traceability but also plays a foundational role in enabling continuous learning.

## 4.4 GNN Feedback Loop

While the RoBERTa classifier proves effective in detecting adversarial prompts in single turn flows it lacks the ability to reason about context or relational patterns over time. To address this, a GNN was introduced as another means of analysis. The GNN models interactions as a directed graph that not only captures each prompt but the temporal flow and semantic relationships across user sessions.

Graphs are constructed with two types of edges, temporal and semantic. Temporal edges simply link sequential prompts within the same session preserving the natural flow of the conversation. Semantic edges are determined by nodes whose cosine similarity exceeds the defined threshold ($\geq$0.89) which is based on the embeddings obtained from the *roberta-base* model, a method consistent with prior graph-based text classification research (Nakajima & S., 2023). Using this set-up, GNNs can correctly detect clear and subtle adversarial patterns, particularly those that span multi-turn dialogue where context is important.

Using PyTorch Geometric, the GNN is implemented using GraphSAGE as the core GNN architecture. GraphSAGE was selected due to its ability to efficiently combine the context of prompts using local relationships without the need for the entire graph to be held in memory. The model is trained to classify each prompt node by leveraging both its embedding and position within the graph. It is then evaluated using standard classification metrics.

## 4.5 Real-Time Hybrid Inference

Once deployed the system operates through a real-time hybrid inference loop that utilises the strengths of both the RoBERTa classifier and the GNN. Each prompt that is inputted is first passed through the RoBERTa classifier, evaluating its potential for adversarial behaviour. If the classifier predicts the prompt is of an adversarial nature with high confidence ($\geq$ 0.98), it is immediately blocked and logged to the database.

If the classifier's confidence score falls between 0.74 and 0.97, it is then passed to the GNN for further analysis. If that analysis produces a probability of adversarial behaviour ($\geq$ 0.92) the prompt is blocked. The GNN operates over a SessionGraph, which tracks user interactions over a single session,

adding prompts as nodes and building both semantic and temporal edges between nodes. This allows the GNN to model evolving context and detect multi-turn adversarial behaviour or obfuscated attack patterns that the RoBERTa classifier may miss.

Whether a prompt is blocked or allowed, the conversation is logged, including the confidence score and the GNN decision by means of a central detection endpoint, including the session ID, prompt, timestamp, classifier score, the GNN threshold band and whether the prompt was blocked or not. While the SessionGraph only exists for the duration of a single session, all logged interactions, are later grouped by session to reconstruct session graphs for GNN retraining. This logging process enables long-term pattern analysis across sessions, helping the system to adapt to evolving adversarial threats over time and any newly flagged prompts can then be added to an adversarial training dataset to improve the classifier and GNN performance.

This hybrid system delivers both real-time response and ongoing adaptability by combining the accuracy of the classifier with the contextual reasoning of the GNN. The system presents a resilient defence against a wide range of adversarial prompt injection attacks and strategies including subtle social engineering and multi-turn attacks.

### 4.5.1 Session Graph Structure and Role

The SessionGraph is a core component of the hybrid inference architecture, enabling the GNN to make informed, contextual aware decisions based on the structure and conversational flow of chatbot interactions. The SessionGraph does not treat prompts individually, it maps the relationships between prompts in a single session, capturing both semantic and temporal patterns which may indicate adversarial intent.

Each node in the graph represents a single prompt submitted by the user. The nodes store metadata including the session ID, prompt, timestamp, classifier score, the GNN threshold band, blocked or not and source. While the graph only persists for the lifetime of the session, it is saved on shutdown in both JSON and PyTorch formats. These saved session graphs are used for retraining of the GNN allowing long-term patterns to be captured across multiple session restarts.

Edges are constructed dynamically based on both temporal and semantic patterns. Temporal edges link prompts in sequence that occur within the same session helping the model understand the flow of the conversation. Semantic edges are formed when prompts exhibit high cosine similarity between their vector representations as computed using the CLS embedding from a RoBERTa model. This captures the relationships between rephrased or semantically similar prompts that may indicate subtle or obfuscated adversarial attacks.

When the RoBERTa classifier does not confidently reject a prompt ($\geq 0.74$ & $< 0.98$), it is added to the current session graph for GNN analysis. It then processes the graph where each node is updated by aggregating information from its neighbours e.g. previous prompts linked semantically or temporally allowing the model to classify the prompt in context.

The structure of the graph allows the GNN to infer when a benign looking prompt may be part of a wider adversarial pattern, for example where it follows several escalating queries or those that may resemble previously flagged attacks. It also supports session level anomaly detection such as sudden shifts in tone or repeated probing within a session.

By maintaining and leveraging a session graph in real-time the system goes beyond traditional classification to support temporal reasoning, prompt linking and adaptive threat detection, all of which contribute to defending against ever evolving, sophisticated prompt injection attacks in customer service chatbots.

## 4.6 Structured Testing Automation

To improve reproducibility and reduce the burden of manual testing, structured testing was carried out using Playwright an open-source framework that enables automated end to end testing of modern web applications. Rather than interacting directly with the Streamlit frontend, direct API calls are sent to the detection endpoints (baseline, classifier-only and hybrid) using Playwright. This approach ensured that every prompt was logged in a consistent manner, along with results and the testing process could be repeated with little input from a human source.

# 5 Evaluation

The effectiveness of the proposed hybrid adversarial prompt detection system across structured real-world scenarios is evaluated in this section. It examines the performance of the baseline and a RoBERTa classifier only configuration and demonstrates the benefits of combining the classifier with a GNN. It also outlines how this system compares to previous work.

## 5.1 Baseline Vulnerability Assessment

The objective of this phase was to assess the robustness of the core chatbot in its most vulnerable state prior to any defence mechanisms being implemented. A test suite of 200 test cases was curated from publicly available adversarial datasets including Tensor Trust, In-The-Wild-Jailbreak and Wild Jailbreak datasets and custom crafted prompts designed to probe for:

- Direct jailbreak attempts
- Indirect prompt injections
- Output leakage e.g. account data exposure
- Instruction hijacking through embedded or multi-layered inputs.

These were submitted directly to the chatbot backend using Playwright, which was configured to send API level requests to the detection endpoints, bypassing the Streamlit frontend. The backend was powered by a Mistral-7B model using Ollama with no classifier, no GNN and a minimal security system prompt. A structured evaluation of this test suite produced the following results:

- Accuracy: 52.50%
- Precision: 100.00%
- Recall: 15.93%
- F1-score: 27.48%

The Recall score here indicates the baseline system, with minimal protection, allowed a significant number of adversarial prompts through only managing to block a small number of them even though it did have perfect Precision when it did act. These results aligned with expectations of a minimally protected systems and highlights the need for more robust, context-aware defences, such as the RoBERTa classifier and session-based GNN integration introduced later.

These results are further illustrated in the confusion matrix (Figure 5.1) and the per-class performance metrics (Figure 5.2). The confusion matrix highlights the high number of false negatives, while the per-class metrics confirm the imbalance between perfect precision and extremely poor recall.

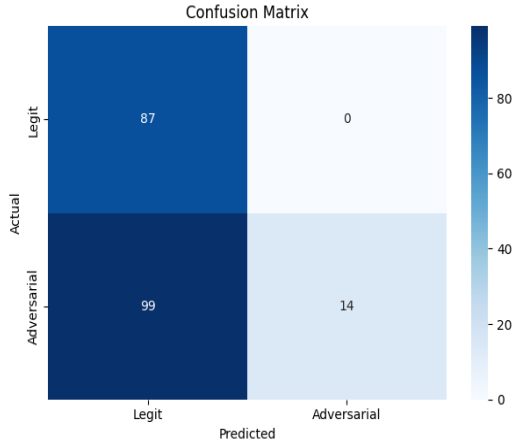


**Figure 5.1 Confusion matrix - baseline chatbot**

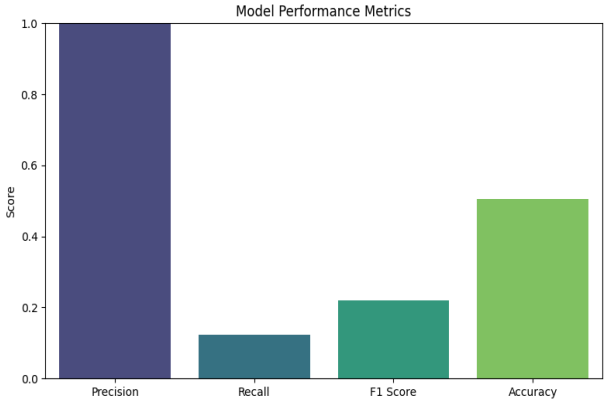


**Figure 5.2 Per-class metrics - baseline chatbot**

This low recall rate in the baseline system aligns with the broader findings in the literature. Pasquini et al. (2024) demonstrated that LLMs without targeted defences were vulnerable to indirect prompt injection attacks in over 70% of tested cases. Similarly, Liu et al. (2024) in *Formalising and Benchmarking Prompt Injection Attacks and Defences* showed that even with modern filters, LLMs were bypassed by crafted jailbreak prompts 60-80% of the time. While this baseline performs slightly better, likely due to the system prompt introducing guardrails and basic keyword filtering, it still importantly demonstrates vulnerability reinforcing the need of context-aware defensive measures.

## 5.2 Classifier Evaluation

The next phase of the evaluation examined the performance of the RoBERTa classifier when used as a standalone mechanism for adversarial prompt detection. This classifier was fine-tuned on a dataset comprised of legitimate and adversarial prompts, where it acted as a first line of defence blocking malicious inputs before they reached the chatbot.

As detailed in section 4.2, the classifier was trained on a task-specific dataset comprising of both legitimate and adversarial prompts. Legitimate samples were aligned to customer service interaction, while adversarial examples included a variety of injection types, including direct and indirect attacks to subtle and obfuscated prompts. Evaluation was conducted using a structured suite of 200 prompts covering the same categories, ensuring consistency and broad coverage in assessing performance.

As noted earlier, classifier training across all epochs was stable indicating strong generalisation without overfitting. The classifier achieved an overall accuracy of 87.00% when run against the structured test suite, correctly classifying 174 prompts. Final evaluation metrics were:

- Accuracy: 87.00%
- Precision: 81.29%
- Recall: 100.00%
- F1-score: 89.68%

Performance across specific categories was notably high. Legitimate prompts were classified with 70.11% accuracy (61/87), while all adversarial prompts were correctly blocked (113/113). The lower precision originated from 26 false positives which were mostly legitimate prompts that were polite, indirect or diverged from the standard syntax or tone.

These results are illustrated in the confusion matrix (Figure 5.3) and the per-class performance metrics (Figure 5.4). The confusion matrix shows all adversarial prompts were correctly identified, while the per-class metrics confirm perfect recall but reduced precision due to false positives on legitimate inputs.

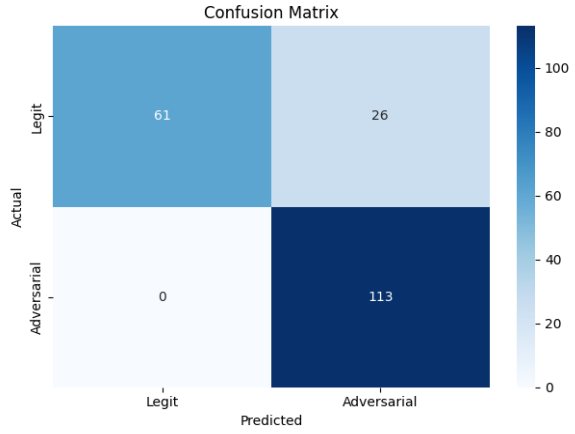


**Figure 5.3 Confusion matrix - classifier only**

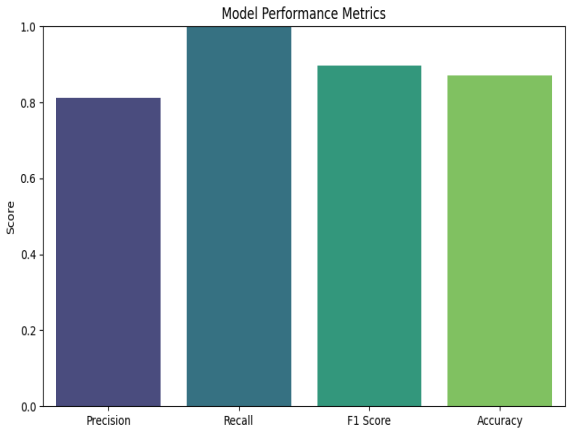


**Figure 5.4 Per-class metrics - classifier only**

The results observed here confirms what was noted within the literature, that is, transformer-based classifiers are highly effective against known attack patterns but remain challenged by subtle or context dependent adversarial inputs (Liu, 2024; Muliarevych, 2024). Notably, the false positives tended to occur on legitimate prompts where the phrasing was overly polite, vague or uncommon syntax was utilised. The use of threshold banding (safe/potentially adversarial/adversarial) helped provide more granular classification. However, it could not mitigate ambiguity around borderline cases, especially in the classifier-only setup where both 'adversarial' and 'potential' labelled prompts resulted in those being blocked.

Compared to the baseline system, where detection accuracy only achieved 52.00%, the classifier marked a substantial improvement in first layer defence achieving 87.00%. This aligns with research findings that highlight the superiority of supervised learning models over rule-based systems in the early stages of prompt screening. However, it also reflects the challenges of treating each prompt in isolation and without taking into consideration the rest of the conversation (Kulkarni et al., 2025).

To address these remaining challenges, especially the inability to evaluate prompts within a broader conversational context, a session-aware Graph Neural Network was introduced. The following section evaluates its impact on multi-turn detection, contextual awareness and real-time adaptability.

## 5.3 Hybrid Detection (RoBERTa Classifier, GNN & Session Graph)

The final phase of the evaluation examined the complete hybrid adversarial detection system, which combines a fine-tuned RoBERTa classifier with a session-aware Graph Neural Network (GNN). This setup was introduced to address the limitations of single-turn detection by incorporating session level context into the inference process. This enabled a more robust handling of ambiguous, obfuscated and multi-turn adversarial techniques.

As outlined in the implementation section, the GNN was trained on a set of 29 session graphs which were constructed from 128 test runs, using the logs stored in a local SQLite database. These logs were processed to construct each session graph using both temporal and semantic relationships between the prompts. This allowed the system to model conversational context and detect adversarial patterns which develops across multiple turns. All session data is now being logged to the database at runtime, and the resulting session graph is constructed and stored at the end of each session. This provides a continuously expanding dataset for any future retraining of the GNN. This approach supports a more adaptive and context-aware detection pipeline capable of evolving with real-world threats.

The system was evaluated using a test suite of 200 prompts, comprising of 113 adversarial and 87 legitimate examples. These included direct, indirect, obfuscated and conversational disguised prompts alongside legitimate queries related to orders, returns and customer support. The same structured test suite was used across the baseline, classifier only and hybrid evaluations to ensure fair comparison. Additionally, real-world testing with BurpSuite was conducted to simulate prompt injection attacks in a live context. These results are discussed in the next section.

The hybrid system achieved an overall accuracy of 95.00%, providing a clear improvement over the classifier only configuration (87.00%). It significantly outperformed the baseline system where detection accuracy only reached 52.00%. Final evaluation metrics for the hybrid system were:

- Accuracy – 95.00%
- Precision – 92.56%
- Recall – 99.12%
- F1-score – 95.73%

The higher precision compared to the classifier-only setup (an increase of ~11%) confirms the hybrid model's ability to reduce false positives, particularly on legitimate prompts with vague phrasing or syntax that deviated from the standard tone. While recall dipped slightly, the addition of session level reasoning allowed the system to detect patterns or evolving adversarial behaviours across turns that had previously evaded detection by operating just below thresholds or by spreading intent across multi-turn dialogues.

These results are further illustrated in the confusion matrix (Figure 5.5) and per-class performance metrics (Figure 5.6). The confusion matrix highlights the reduction in false positives compared to the classifier-only system, while the per-class metrics show a strong balance between high recall and improved precision.

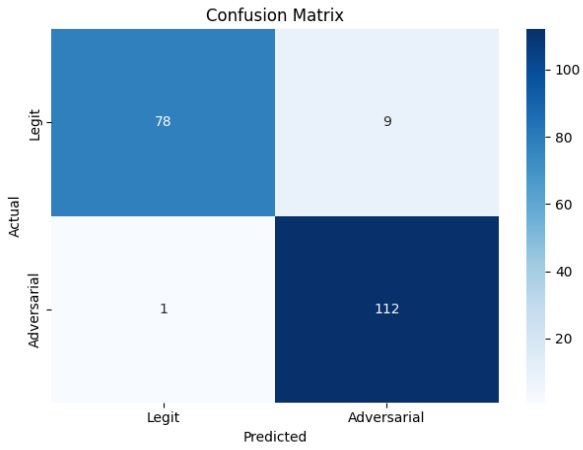


**Figure 5.5 Confusion matrix - hybrid system**

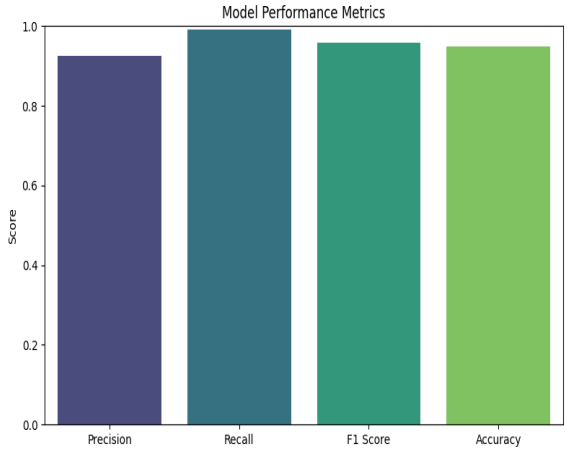


**Figure 5.6 Per-class metrics - hybrid system**

As noted, the baseline system achieved an accuracy rate of 52.00% while the hybrid system detection accuracy reached 95.00%, a significant improvement. This aligns with the recent research encouraging layered defences that integrate both local and contextual decision making (Pasquini et al., 2024; Liu, 2024). The ability to flag subtle prompt injection patterns based on repurposing legitimate-looking phrases or wording for malicious intent, conversational role shifts or edge cases that don't fit normal attack patterns highlights the strengths of combining transformer based and graph-based reasoning.

Together, these results demonstrate that the hybrid detection system represents a meaningful advancement over both the baseline and standalone classifier methods. By unifying prompt level classification with session-level context the system achieves a more accurate, resilient and future proofed defence against prompt injection and output leakage in customer service chatbots.

## 5.4 Real-world Testing

To validate the strength of the hybrid adversarial detection system under realistic deployment conditions, a structured real-world testing phase was undertaken. This included both direct interaction with a chatbot interface and adversarial simulation using Man-In-The-Middle (MITM) interception techniques to manipulate inputs during transit, simulating a hostile environment beyond normal user misuse.

The real-world testing environment comprised a Streamlit based chatbot frontend through which all prompts were routed through the complete detection pipeline with responses returned in real-time. To streamline evaluation and adversarial testing workflows, the frontend included a toggle mechanism via checkboxes to allow the user to seamlessly switch between standard real-world testing and a BurpSuite driven MITM mode. When the BurpSuite mode was enabled all outgoing requests from the frontend were rerouted through a local proxy configured for MITM interception allowing for adversarial payloads to be injected into otherwise legitimate prompts.

To facilitate this adversarial simulation, BurpSuite was configured as a local proxy. This was achieved through implementing a system wide proxy (127.0.0.1:8080) and its SSL certificate was added to the local trust store to enable encrypted traffic detection. Prompts could then be intercepted using the Intruder tool within BurpSuite, which were then modified in real-time through techniques

like subtle modifications to legitimate prompts, indirect prompt injections attempting to elicit sensitive information and multi-turn escalation tactics before being forwarded to the backend detection API for evaluation. By testing at this level, the evaluation extended beyond simple prompt-based attacks and examined the system's resilience to more evasive techniques aimed at bypassing filters or frontend safeguards.

Final evaluation compared the system's actual decisions against the expected outcome, whether a prompt was blocked or not. At the test case level, multi-turn adversarial cases were treated as single logical failures, and the hybrid detection system achieved the following:

- Accuracy – 95.00%
- Precision – 90.91%
- Recall – 100.00%
- F1-score – 94.87%

These results reflect strong real-world performance, with only four legitimate prompts incorrectly blocked (false positives) and no adversarial prompts incorrectly allowed (false negatives). The system reliably identified prompt injection attempts despite obfuscation, context manipulation or subtle rephrasing of prompts, demonstrating in live conditions, its effectiveness. The inclusion of the MITM attacks using BurpSuite greatly improved the realism of the evaluation and confirmed the system's robustness to request level tampering beyond controlled test suites.

## 5.5 Discussion

The evaluation results highlight the effectiveness of the hybrid adversarial prompt detection system in mitigating a wide range of attack strategies across test scenarios that are both structured and rooted in the real-world. By implementing a fine-tuned RoBERTa classifier alongside a session-aware Graph Neural Network (GNN), the system consistently outperformed the baseline configurations and showed meaningful improvement over the classifier only system establishing robust detection capabilities across direct, indirect, obfuscated and multi-turn prompt types.

The initial evaluation of the baseline outlined the vulnerability of the chatbot. Operating solely on a minimal system prompt with a small adversarial keyword list and no classifier or GNN, the system failed to block a significant number of prompts that were adversarial. These included a wide range of attack types such as direct, indirect, obfuscated and role-subverting prompts. Some of these types of prompts, while not clearly tailored to customer service contexts, proved effective at bypassing weak security measures. This illustrates the wider range of threats that contemporary chatbots with LLM capabilities have to deal with. The findings mirror those of Pasquini et al. (2024) and Liu et al. (2024) who demonstrated high bypass rates in those systems that lack context-aware evaluation mechanisms.

In the classifier only configuration, the introduction of a fine-tuned RoBERTa classifier improved detection performance significantly. Accuracy rose to 87.00% with perfect Recall (100.00%) but lower Precision (81.29%) due to false positives on legitimate prompts. The classifier excelled at detecting well defined adversarial behaviours, including many crafted prompts from non-domain sources that leveraged common jailbreak strategies or indirect prompt injection techniques. However, its performance declined on legitimate prompts that deviated from standard syntax, tone or intent

often misclassifying these as potential threats. Some false negatives arose from legitimate cases that utilised polite, indirect or unusual phrasing, while false negatives were not observed due to perfect Recall. Although the introduction of threshold banding helped as it introduced more granularity, these borderline cases remained difficult to classify without access to conversational context.

The hybrid detection system incorporated the fine-tuned RoBERTa classifier and the GNN which addressed these limitations by introducing session level reasoning through the use of temporal and semantic graph structures. The configuration achieved an accuracy rate of 95.00% in structured testing and maintained strong performance (95.00%) during real-world testing. The hybrid system showed a marked improvement in precision (92.56%, up ~11% from classifier-only), while recall dipped slightly. This trade-off reduced false positives substantially while still detecting the majority of adversarial prompts. By modelling interactions across entire sessions, the GNN enabled the system to detect patterns that would otherwise escape prompt-level analysis.

At the end of each session, the session graphs were all saved, forming the foundation of an active feedback loop. While the retraining of the GNN is not automated and remains a manual step, the automated capture and storage of the session data eliminate the need for manual log reconstruction. This allows for iterative improvement over time, allowing the system to adapt to emerging adversarial threats and as such, ensures its long-term resilience.

The hybrid model's strength lies in its improved precision and accuracy, demonstrated in its F1-score (95.73%) which reflects an effective trade off between detecting those prompts which are adversarial in nature and preserving legitimate user inputs. It's important to note that the hybrid system reduced the number of false positives on ambiguous but benign prompts, an area where the classifier only configuration had struggled. In a customer service setting, this is particularly relevant, as false blocks degrade user experience and false negatives pose a security risk.

Despite the strong performance in both structured and real-world evaluations, certain limitations persist. Compared to the classifier only setup, the hybrid system did manage to reduce the number of false positives. Occasional misclassification still occurred, particularly when handling prompts which were legitimate but that contained unusual phrasing or indirect requests with syntax that deviated from the standard norm. Although the GNN enhanced detection of multi-turn and obfuscated adversarial attacks, its effectiveness is ultimately bound by the diversity and quality of session graphs it is retrained on. Infrequent or novel conversational patterns may escape detection until such a time that sufficient examples have been captured, logged and incorporated into any future training cycles.

Overall, the consistent results observed across all evaluation phases, including structured test suites and real-world Man-In-The-Middle simulations demonstrate the hybrid system's advantage over more traditional single layer defence systems. Incorporating adversarial intent from a variety of perspectives, including prompts that go beyond standard customer misuse, yielded more depth in the evaluation. The system's performance against these types of attacks strengthens its ability to generalise effectively, ensuring that when it is confronted with novel or out of context attacks it demonstrates continued resistance.

# 6 Conclusion and Future Work

This research set out to answer the question '*How effective is a defensive AI model in detecting and mitigating adversarial prompt injection attacks in customer service chatbots?*' The objectives of this research included the design and implementation of a hybrid adversarial detection system comprising a fine-tuned RoBERTa classifier alongside a Graph Neural Network (GNN) operating with an active feedback loop. Through structured and real-world testing, the system proved highly successful in achieving this goal.

Key aspects of the system developed included the hybrid detection pipeline, the implementation of a logging and feedback mechanism for continuous learning and the integration of session graphs to enable context-aware detection. Evaluation showed that the system outperformed the baseline setup and detailed a clear improvement over the classifier only configuration achieving 95.00% accuracy in structured tests and in real-world Man-In-The-Middle (MITM) simulations. These results demonstrate the effectiveness of a combined prompt level and session level reasoning to detect obvious and subtle adversarial behaviours.

While the system performed well, it is not without its limitations. False positives did occur on legitimate ambiguous prompts, or unusually rare adversarial strategies may go undetected until such time they have been captured in future session graphs and the GNN retrained accordingly. Also worth noting, in its current state, the GNN retraining process remains a manual task even though prompt logging is automatic

As a proof of concept, the system requires future development to leverage its full potential for scalable deployment. Focusing future efforts on automated retraining and enhanced adaptability to emerging and evolving threats, extending the system across new chatbot domains to test its adaptability, and integrating explainability features into the GNN to improve transparency and trust in the classifications ensures the evolution and effectiveness of the system.

Ultimately this research provides a strong basis for adaptive, real-time adversarial prompt detection in LLM-driven customer service chatbots, offering a viable foundation for building more secure, context-aware conversational systems that can evolve alongside the threats they face.

# 7 References


Bahar, A. F. K. M. M. S. H. a. A. K., 2025. CONTINUUM: Detecting APT Attacks through Spatial-Temporal Graph Neural Networks. *arXiv preprint arXiv:2501.02981.*, 06 01.

Hedyeh Nazari, A. Y. A. D. F. Z. G. S., 2024. P3GNN: A Privacy-Preserving Provenance Graph-Based Model for APT Detection in Software Defined Networking. *arXiv preprint arXiv:2406.12003.*, 24 06.

Iqbal, F. &. S. F. &. K. F. &. M. Á., 2023. When ChatGPT goes rogue: exploring the potential cybersecurity threats of AI-powered conversational chatbots.. *Frontiers in Communications and Networks.,* Volume 4.

Jiahao Yu, Y. W. D. S. M. J. S. Y. X. X., 2023. Assessing Prompt Injection Risks in 200+ Custom GPTs. *arXiv preprint arXiv:2311.11538.*, 20 11.

Jingwei Yi, Y. X. B. Z. E. K. G. S. X. X. a. F. W., 2023. Benchmarking and Defending Against Indirect Prompt Injection Attacks on Large Language Models. *arXiv:2312.14197*, 23 12.

Jones, N., Whaiduzzaman, M. & Jan, T., 2025. A CIA Triad-Based Taxonomy of Prompt Attacks on Large Language Models.. *Future Internet,* 17(3).

Liu, Y. J. Y. G. R. J. J. a. G. N., 2024. Formalizing and benchmarking prompt injection attacks and defenses. *arXiv preprint arXiv:2310.12815v4.*

M. Charfeddine, H. M. K. B. H. a. M. G., 2024. ChatGPT's Security Risks and Benefits: Offensive and Defensive Use-Cases, Mitigation Measures, and Future Implications. *IEEE Access,* Volume 12, pp. 30263-30310.

Mohammadian, H. H. G. A. S. R.-F. R. a. G. A., 2024. Explainable Malware Detection through Integrated Graph Reduction and Learning Techniques.. *arXiv preprint arXiv:2412.03634.*, 04 12.

Muliarevych, O., 2024. *Enhancing system security: LLM-driven defense against prompt injection vulnerabilities.* s.l., s.n., pp. 420-423.

Nakajima, H. &. S. M., 2023. Text Classification Based on the Heterogeneous Graph Considering the Relationships between Documents.. *Big Data and Cognitive Computing,* 6(4), p. 181.

Pasquini, D. K. E. a. A. G., 2024. Hacking Back the AI-Hacker: Prompt Injection as a Defense Against LLM-driven Cyberattacks.. *arXiv preprint arXiv:2410.20911*, 28 10.

Prashant Kulkarni, A. N., 2025. Temporal Context-awareness: A Defense Framework Against Multi-turn Manipulation Attacks on Large Language Models. *arXiv:2503.15560*, 18 03.

Qiao, W. F. Y. L. T. M. Z. S. Y. M. J. a. L. Y., 2024. Slot: Provenance-Driven APT Detection through Graph Reinforcement Learning.. *rXiv preprint arXiv:2410.17910.*, 23 10.

Rossi, S. M. A. M. R. a. T. J., 2024. An early categorization of prompt injection attacks on large language models. *arXiv preprint arXiv:2402.00898.*, 31 01.

Tshimula, J. N. X. N. D. T. P. K. F. F. M. a. W. S., 2024. Preventing Jailbreak Prompts as Malicious Tools for Cybercriminals: A Cyber Defense Perspective.. *arXiv preprint arXiv:2411.16642*, 25 11.

Zhong, M.-H. &. L. M. &. Z. C. &. X. Z., 2024. A Survey on Graph Neural Networks for Intrusion Detection Systems: Methods, Trends and Challenges. Computers & Security.. *Computers & Security.,* Volume 141.

# 8 Appendix A

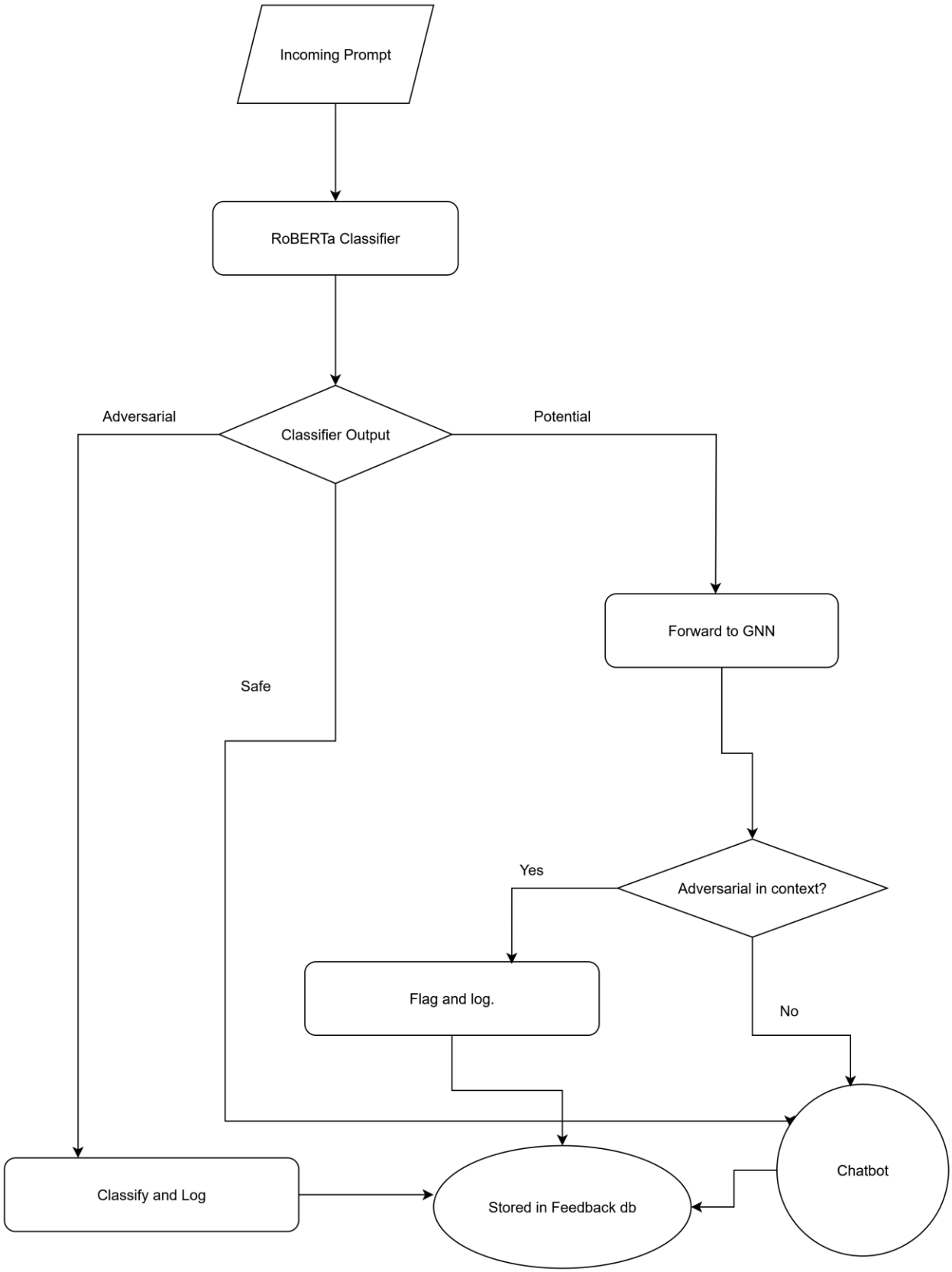


Figure 1: Workflow Diagram for Adversarial Defence and Mitigation